CLAIMS

What is claimed is:

1. A method for automatically generating source code for manipulating at least one mark-up language message based on a mark-up language message definition comprising the steps of:

a) receiving the mark-up language message definition;

b) generating a first in-memory representation of the message definition based on the received message definition; and

c) generating a second in-memory representation of source code based on the first in-memory representation of the message definition.

2. The method of claim 1 further comprising the step of:

d) generating source files based on the second in-memory definition of the source code.

3. The method of claim 1 wherein the first in-memory representation is a schema object tree corresponding to an XML Schema message definition; wherein the schema object tree includes one or more nodes.

4. The method of claim 1 wherein the second in-memory representation includes one of class members, class methods, source file object nodes, class object nodes, and source file comment object nodes.

5. The method of claim 2 wherein the step of generating the second in-memory representation of source code based on the in-memory representation of the message definition includes the step of

generating a source object tree by employing a blackboard architecture that includes agents and solutions; wherein the source object tree includes one or more nodes; and wherein the nodes of the schema object tree are agents and the nodes of the source object tree are the solutions.

6. The method of claim 2 wherein the second in-memory representation includes elements and attributes; wherein the step of generating source files based on the second in-memory representation of the source code includes the step of

writing the elements and the attributes into respective Java class source files.

7. The method of claim 5 wherein the step of generating a source object tree by employing a blackboard architecture includes the step of

performing context sensitive compilation while generating each node of the source object tree.

8. The method of claim 7 wherein the step of performing context sensitive compilation while generating each node of the source object tree includes

performing pre-fix processing.

9. The method of claim 7 wherein the step of performing context sensitive compilation while generating each node of the source object tree includes

performing in-fix processing.

10. The method of claim 7 wherein the step of performing context sensitive compilation while generating each node of the source object tree includes

performing post-fix processing.

11. The method of claim 1 wherein the mark-up language is XML.

12. The method of claim 1 wherein the mark-up language message definition is an XML schema message definition.

13. The method of claim 2 wherein the source code stores information included in at least one XML message.

14. The method of claim 2 wherein the source code manipulates information included in at least one XML message.

15. The method of claim 16 wherein the method generates a communication API based on an XML schema definition.

16. The method of claim 1 wherein the method automatically parses context sensitive grammar in the compilation of XML schema to source code.

17. A system for generating source code for manipulating at least one mark-up language message comprising:

a) a first module for receiving a message definition and based thereon for generating a first in memory data structure that corresponds to the message definition; and

b) a second module for receiving the first data structure and based thereon for generating a second in memory data structure that corresponds to source code for manipulating at least one mark-up language message.

18. The system of claim 17 wherein the first data structure includes a plurality of nodes and the second data structure includes a plurality of nodes, the system further including:

a blackboard architecture; wherein the nodes of the first data structure are agents and the nodes of the second data structure are solutions.

19. The system of claim 17 further comprising:

a mechanism for handling context sensitive grammar; wherein the processing for a current node in the first data structure considers child nodes of the current node and the parent node of the current node.

20. The system of claim 17 wherein the source code includes Java class source files.

21. The system of claim 17 wherein the mark-up language message is an XML mark-up language message.

22. A method for automatically generating source code for manipulating at least one mark-up language message comprising the steps of:

a) receiving a schema definition for a mark-up language message;

b) generating a first in-memory representation of the schema definition based on the schema definition;

c) generating a second in-memory representation of source code based on the first in-memory representation of the schema definition; wherein the step of generating a second in-memory representation of source code based on the first in-memory representation of the schema definition includes

performing one of context free processing and context sensitive processing.

23. The method of claim 22 further comprising the steps of:

generating one or more source code files based on the second in-memory representation of source code.

24. The method of claim 22 further comprising:

reading a portion of a schema definition that corresponds to one or an element or an attribute from a schema definition file;

constructing a schema object hierarchy based on the read portion; and

compiling the object hierarchy into a source object hierarchy; and

writing the source object hierarchy to one or more object-oriented source files.

25. The method of claim 24 wherein schema object hierarchy includes a plurality of objects; wherein each object includes code to compile itself into a source code primitive.

26. The method of claim 24 wherein the source object hierarchy includes a special set of objects that represent a predetermined class source file and that has a predetermined number of members, methods and definitions.

27. The method of claim 24 wherein the source object hierarchy includes an object corresponding to a whole source file, an object corresponding to a file declaration comment, an object corresponding to a package name, an object corresponding to import statements, and object corresponding to class definitions.

28. The method of claim 27 wherein the object for class definition includes one of an object corresponding to declaration statement, an object

corresponding to specific class member definition, and an object corresponding to method definition.

29. The method of claim 24 wherein each source object is programmed to write itself into a respective source file.

30. The method of claim 29 wherein each source object includes a toString() method that recursively calls toString() method of its descendents to write itself into a respective source file.